July 2008

KNOWLEDGE ACQUISITION WITH SUPERVISED ONTOLOGY POPULATION

kaiquan xu  
*City University of Hong Kong*, kaiquan.xu@student.cityu.edu.hk

Stephen Shaoyi Liao  
*City University of Hong Kong*, issliao@cityu.edu.hk

Raymond Y.K. Lau  
*City University of Hong Kong*, raylau@cityu.edu.hk

Lejian Liao  
*Beijing Institute of Technology*, liaolj@bit.edu.cn

Follow this and additional works at: [http://aisel.aisnet.org/pacis2008](http://aisel.aisnet.org/pacis2008)

**Recommended Citation**  
[http://aisel.aisnet.org/pacis2008/85](http://aisel.aisnet.org/pacis2008/85)

This material is brought to you by the Pacific Asia Conference on Information Systems (PACIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in PACIS 2008 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
KNOWLEDGE ACQUISITION WITH SUPERVISED ONTOLOGY POPULATION

Kaiquan Xu, Department of Information System, City University of Hong Kong, Hong Kong, China, kaiquan.xu@student.cityu.edu.hk

Stephen Shaoyi Liao, Department of Information System, City University of Hong Kong, Hong Kong, China, issliao@cityu.edu.hk

Raymond Y.K. Lau, Department of Information System, City University of Hong Kong, Hong Kong, raylau@cityu.edu.hk

Lejian Liao, Department of Computer Science and Engineering, Beijing Institute of Technology, Beijing, China, liaolj@bit.edu.cn

Abstract

Ontology plays a crucial role in capturing and disseminating business information (e.g., products, services, relationships of businesses) for effective human computer interactions. However, manual construction of domain ontology is very labour intensive and time consuming. This paper illustrates a novel ontology population method for semi-automatic business knowledge acquisition from text. In particular, the proposed method is underpinned by the effective SVM-struct algorithm which treats ontology population as a sequence labelling problem. By automatically exploring taxonomy knowledge captured in domain ontology, our ontology population method can effectively classify objects to multiple categories. The initial experimental results show that the SVM-struct based ontology population method which utilizes taxonomy knowledge outperforms other traditional methods in a benchmark ontology population task.

Keywords: Knowledge Management, Domain Ontology, Ontology Population, Data Mining, Machine Learning.
INTRODUCTION

Knowledge is the most important corporate asset and it is the key for organizations to achieve sustainable competitive advantage. Knowledge management is a collection of processes that govern the creation, dissemination, and utilization of knowledge (I. Nonaka et al., 1995). To be able to effectively manage the intellectual capital, organizations need an effective mechanism to identify and capture information and knowledge about business processes, products, services, markets, customers, suppliers, and competitors to improve the organizations' goal achievement. Ontology allow domain knowledge such as products, services, markets, etc. to be captured in an explicit and formal way such that it can be shared among human and computer systems (Taehee Lee et al., 2006).

Ontology often refers to a formal specification of conceptualization (T. R. Gruber et al., 1993). In other words, ontology is a formal representation of concepts and their interrelationships. The notion of ontology is becoming very useful in various fields such as intelligent information extraction and retrieval, semantic Web, electronic commerce, and knowledge management (C. A. Welty et al., 2003, R.Y.K. Lau, 2007). Although ontology is useful in many areas, engineering ontology turns out to be very expensive and time consuming. Automated ontology extraction is vital for the success of ontology applications because it alleviates the classical problem of knowledge acquisition bottleneck. Ontology population is one of the sub-tasks of ontology extraction and it refers to the extraction of the instances of concepts, relations or attributes in ontology (H.G. Yoon et al., 2007) (here we only consider the extraction of instances of concepts). Figure 1 shows an example of populated (classified) instances under the automotive domain ontology. For instance, the car instances such as Toyota camry, Toyota corolla, Mazda 626 etc. are populated under the family car category. Domain ontology with populated instance is very useful for many business applications. For example, for the recommender system in the context of e-Commerce, ontology population can automatically populate various products and services to the related classes to improve the precision of recommendations. Take an example, the recommender system can automatically recommend Toyota camry to a consumer if the system knows that the consumer is looking for family car.

![Figure 1. Ontology Population for an Automotive Domain Ontology.](image-url)

In this paper, we propose a novel supervised machine learning approach for automatic ontology population from text with the combination of sequence model and maximum margin approach. The previous work on ontology population often adopted unsupervised learning approaches, and the biggest advantage of unsupervised methods is that they do not require the costly annotated training
examples. But there is a serious shortage of the supervised ones: low recall, they can only extract some obvious instances, and large numbers of hidden instances are ignored. So now some supervised methods (Jun’ichi Kazama et al., 2002) are explored with the annotated corpus (e.g., the Genia corpus) for this problem. Existing supervised ontology population methods adopt classical classification algorithms (such as SVM) to populate instances (Jun’ichi Kazama et al., 2002). In fact, ontology population, very similar with Name Entity Detection (NED) (Daniel M. Bikel et al., 1999), involves in two problems: one is finding the bounds of instances, and another is identifying the classes of instances. For finding the bounds of instances, sequence labelling models are powerful, as sequence models can capture the relationships between labels. In our approach, to conduct sequence labelling, the structured Support Vector Machine (SVM-struct) (Nam Nguyen et al. 2007) algorithm with maximum margin principle is adopted to learn the mapping function between input sentences and output labels, and the similar Hidden Markov Model (HMM) (Lawrence R et al. 1998) is used as discriminant function. Through this way, the advantages of sequence labelling model (HMM) and maximum margin classifier are combined for ontology population.

Another common weakness of existing supervised ontology population methods is that they do not exploit the taxonomy relations in domain ontology for instance population (Jun’ichi Kazama et al., 2002). Nevertheless, (A. McCallum et al. 1998, A. S. Weigend et al., 1999, Lijuan Cai et al., 2004) point out that taxonomy knowledge can bootstrap classification performance, particularly when the number of training examples is relatively small. One contribution of our research work is the development of a supervised ontology population method which can utilize the taxonomy knowledge to improve classification performance even if only a small number of training examples and features are available. By utilizing taxonomy information captured in ontology, the proposed SVM-struct algorithm based ontology population method offers powerful generalization capability for classifying instances. Since real-world application domains are very complex, an instance is typically associated with more than one class. For example, a car can be classified as both a family car and a sport car (e.g., Mercedes-Benz CLK350). Existing supervised ontology population methods fail to address the multi-label problem (Jun’ichi Kazama et al., 2002). Our method also involves a little work to support such complex real-world business applications.

The rest of the paper is organized as follows. Section 2 describes the related research in ontology population. Section 3 takes the ontology population as sequence labelling, and provides the computational mechanisms via three discriminant functions. Section 4 describes the experimental procedures and reports our initial results. Finally, we offer concluding remarks and describe future direction of our research work.

2 RELATED WORK

2.1 Ontology Population

The unsupervised methods can be divided into two categories: one utilizes patterns (Hearst, Marti 1998, P. Velardi et al., 2005), and the other utilizes contextual features (P. Cimiano et al., 2005). Pattern-based approaches search for some patterns such as "is-a" between two terms to extract the instances and classes. (P. Velardi et al., 2005) used head-matching heuristics to identify the “is-a” relation. For example, if a term t1 is the head of the term t2, t2 is a kind of t1. Although this method leads to good accuracy result, its recall is relatively low because these kinds of patterns do not appear frequently in a text corpus. The contextual features based approaches extract the features describing classes from a corpus; the features of instances are then compared with that of a class to decide class membership. Contextual features can be superficial (M. Fleischman et al., 2002) or syntactic (A. Almuhareb et al., 2004). Ontology population was also explored in the context of news summarization (Chang-Shing Lee et al., 2005). Domain ontology with various categories of news events was manually constructed by domain experts. A fuzzy inference mechanism was developed to generate the
membership degrees for each event (terms) extracted from online news with respect to the fuzzy concepts defined in the fuzzy ontology. A standard triangular membership function was used for the classification purpose. In general, unsupervised ontology population methods do not lead to good classification performance although the advantage of these methods is that no training data is required.

The supervised ontology population methods usually need some annotated training data. In (M. Fleischman et al., 2002), word frequency, topic signature, and WordNet features are used to classify terms to various categories using different learning algorithms. In (Jun'ichi Kazama et al., 2002), ontology population was treated as a labelling problem, and they used multi-classes SVMs to classify the terms extracted from a text corpus. However, the multi-classes SVMs did not exploit taxonomy information and they could not classify a single term to more than one class.

2.2 The SVM-struct Algorithm

The SVM-struct algorithm (Ioannis Tsochantaridis et al., 2005, Ioannis Tsochantaridis et al., 2004) uses the large margin method from SVM to predict structured outputs, such as tree, sequence, and sets. The previous experiments show that SVM-struct has much better performance than Hidden Markov Model (HMM), Conditional Random Fields (CRF), etc (Nam Nguyen et al. 2007). The basic idea of SVM-struct is as follows:

Let \( \{(x_i, y_i)\}_{i=1}^{n} \) be a set of \( n \) labeled training instances, \( y_i \) can be any complex structure; a linear discriminant Function is defined by \( F(x, y; \omega) = \langle \omega, \Psi(x, y) \rangle \). So, the corresponding classification function \( f \) is defined by \( f(x; \omega) = \arg \max_{y \in Y} F(x, y; \omega) \). The margin of a weight vector \( \omega \) with respect to an instance \( (x_i, y_i) \) can be defined as \( \gamma_i(\omega) = F(x_i, y_i; \omega) - \max_{y \neq y_i} F(x_i, y, \omega) \). We can apply the maximum margin principle to determine the weight vector \( \omega^* \) for achieving optimal separation, that is, \( \omega^* = \arg \min_{\|\omega\|_1} \max_{i=1}^{n} \gamma_i(\omega) \). The key of SVM-struct is to find a right discriminant function \( F(x, y; \omega) \) to reflect the relations between input and output, and deduce the output space size given that fact that the structured output space usually is huge.

3 SUPERVISED ONTOLOGY POPULATION

3.1 Ontology Population as Sequence Labelling

We adopt the approach from Named Entity Detection (NED) (Daniel M. Bikel et al., 1999) to model supervised ontology population as a sequence labeling problem which is supported by the SVM-struct algorithm. Previous work treated NED as the classification task by representing coded region information. For example, in the BIO representation, the region information is coded by the prefixes “B-”, “I-”, and the class “O”. B- means that the current word is the beginning of a named entity, I- means that the current word is in a named entity, and O indicates that the word is not found in a named entity. The following is an example based on the Genia corpus:

“In primary T lymphocytes we show that CD28 ligation leads to…”.

cell_type protein_molecule

“In primary T lymphocytes we show that CD28 ligation leads to…”

O B-cell_type I-cell_type I-cell_type O O O B-protein_molecule O O O...
Accordingly, we adopt the BIO method to model ontology population as a sequence labeling problem as follows. For a given the ontology $Ont$, the class set is defined as $\text{ClassSet} = \{c_i | c_i \in Ont\}$. The predecessor relation is defined as $\prec \text{ClassSet} \rightarrow \text{ClassSet}$, $c_i \prec c_j$ if $c_i$ is the super class of $c_j$ in the ontology. The leaf node class set is defined as $\text{LeafClassSet} = \{c_i | \exists c_j \in \text{ClassSet} \land c_j \prec c_i\}$. The internal node class set is defined as $\text{InterClassSet} = \{c_i | \exists c_j \in \text{ClassSet} \land c_j \prec c_i\}$. Then, the label set from this ontology is defined as $\text{LabelSet} = \{\text{"B\_"}, \text{"I\_"}, \text{"O"}\}$, that is, $\text{LabelSet}$ consists of leaf node classes with labels “B_” or “I_” to indicate the beginning or internal of the class.

The supervised ontology population problem is defined as follows: the input space of ontology population is: $X = \{x_i | x_i = w_1w_2...w_n, w_i \in \text{WordSet}\}$ whereas $x_i$ is a sentence consisting of multiple words; the output space is defined by: $Y = \{y_i | y_i = l_1l_2...l_n, l_i \in \text{LabelSet}\}$, whereas $y_i$ is the label sequence to indicate the scope of the class instance in the input sentence.

Given some training samples: $\{(x_i, y_i)\}_{i=1}^n$, here $x_i \in X, y_i \in Y$, that is some sentences and the label sequences, which indicate the classes of some words, a mapping function $f : X \rightarrow Y$ is learnt using a machine learning algorithm. For a new sentence $x$, $f(x)$ is the prediction of the corresponding label sequence, which indicates some instances belonging to certain classes. Obviously, the output of the mapping function $f$ is the structured sentence, so the SVM-struct approach is the right method for this problem. Since the linear discriminant function $F(x_i, y_i; \omega)$ plays a key role in this approach, three discriminant functions are examined in this paper.

### 3.2 Basic Discriminant Function

For the structured output, a major concern is that one input may have a very large number of output options, which lead to intractable result, particularly if traditional multi-class approaches are applied. For example, for the sentence $x_i$ with the length $|x_i|$, there will be $|\text{LabelSet}|^{|x_i|}$ optional label sequences. For resolving this difficulty, the idea of HMM (Lawrence et al. 1998) can be used. $\Psi(x_i, y_i)$ can be defined as some features describing the label transitions and some features describing the relations between inputs and labels, then the dynamic programming, Viterbi algorithm (Lawrence et al. 1998) can be used to efficiently find $y_i$ to maximize $F(x_i, y_i; \omega)$. If only first order label transition is considered, the discriminant function can be defined by:

$$F_{\text{basic}}(x_i, y_i; \omega) = \sum_{t=1}^{l_i} \sum_{l \in \text{LabelSet}} \langle \Phi(w_i), \delta(l_i, l) \rangle + \eta \sum_{t=2}^{l_i} \sum_{l \in \text{LabelSet}} \sum_{\hat{l} \in \text{LabelSet}} \tilde{\omega}(l, \hat{l}) \delta(l_{t-1}, l) \delta(l_t, \hat{l})$$

where $\Phi(w_i)$ is the features of the $i$-th word in $x_i$, such as the preceding and the following words, prefix, suffix, POS tag, modifier, subject, object etc. $\delta(l_i, l) = \begin{cases} 1 & l_i = l \\ 0 & l_i \neq l \end{cases}$ which is an indicator function. $\eta$ is a scaling factor to balance the two types of feature weight.
3.3 Discriminant Function with Taxonomy Information

For document categorization problem, taxonomy information plays an important role in generalization. Ontology population is also a generalizing process for terms. Therefore, applying taxonomy information to ontology population should improve the performance in general. In order to do this, some features need to be added to describe the relationships between an input term and the predecessor classes rather than purely the leaf classes. So, the discriminant function can be defined as:

\[
F_{\text{hierarchical}}(x_i, y; \omega) = \sum_{i=1}^{k} \sum_{l \in \text{LabelSet}} \langle \sigma_j, \Phi(w_i) \rangle \lambda(l, l) + \sum_{i=1}^{k} \sum_{l \in \text{LabelSet}} \omega(\bar{j}, \bar{i}) \delta(l, l) + \sum_{i=1}^{k} \sum_{l \in \text{LabelSet}} \omega(\bar{j}, \bar{i}) \delta(l, l)
\]

where \( \lambda(l, l) \) is used to indicate the predecessor relation between \( l_i \) and \( l \). For example, \( \lambda(l, l) \) can be designed as:

\[
\lambda(l, l) = \begin{cases} 
1 & \text{if } l = l_i \\
1/p & \text{if } l < l_i \\
0 & \text{otherwise}
\end{cases}
\]

Here \( p \) is the path length from \( l_i \) to \( l \) with reference to the taxonomy relations of the ontology; through this function, the taxonomy information can be integrated into the discriminant function.

3.4 Discriminant Function for Multi-Label Problem

For multi-label problem, e.g., one term with more than one class label, the label sequence \( y_i \) becomes \( y_i = l_i l_i' \) (assuming not more than two labels here), and then the discriminant function becomes:

\[
F_{\text{multi-label}}(x_i, y; \omega) = \sum_{i=1}^{k} \sum_{l \in \text{LabelSet}} \langle \sigma_j, \Phi(w_i) \rangle \lambda(l, l) + \sum_{i=1}^{k} \sum_{l \in \text{LabelSet}} \omega(\bar{j}, \bar{i}) \delta(l, l) + \sum_{i=1}^{k} \sum_{l \in \text{LabelSet}} \omega(\bar{j}, \bar{i}) \delta(l, l)
\]

Since the discriminant function needs to use a combination of labels, it will produce more optional label sequences and may lead to the degradation of computational efficiency. Determining the right number of class labels assigned to a term is a big challenge.

4 EXPERIMENTAL EVALUATION

4.1 Experiment Design

In order to evaluate the performance of our approach for supervised ontology population, a controlled experiment was performed. The results are compared with that of the traditional multi-class SVM classification methods.

The GENIA benchmark corpus (T. Ohta et al., 2002) was used for this experiment. The GENIA corpus is an annotated corpus for the biology domain. In other words, some terms are annotated with the GENIA class labels. The ontology for this experiment comes from the organic branch of the GENIA ontology and it contains 27 classes with 4 levels and 23 leaf node classes. After standard text preprocessing, 1144 sentences and the related label sequences were obtained. Every word is associated with two types of features: 1) word feature \( w_k : 1 \quad (k = -2,-1,0,1,2) \), \( k \) is the relative position from the
current word, and a negative value representing the preceding word, a positive value representing the following words. \( w_i \) is the unique id of a particular word; 2) Part-Of-Speech (POS) tag of the current word \( \text{pos} : 1 \), \( \text{pos} \) is the unique id for the pos tag. The first order label transition was considered in this experiment.

The 1144 sentences were split into two groups: 586 (about 16,000 words) as training examples, 558 (about 15,000 words) as testing examples. The discriminant functions \( F_{\text{basic}}(x_i, y_i; \omega) \) and \( F_{\text{hierarchical}}(x_i, y_i; \omega) \) were applied separately in this experiment. The three performance metrics widely used in the information retrieval community, namely, precision, recall, and F-score were adopted in this experiment. Precision means the ratio of the right labeled instances to all labeled instances by the algorithm, recall means the ratio of the right labeled instances by the algorithm to all original instances, and f-score is a mixture of precision and recall \( (f - \text{score} = \frac{2 \cdot \text{precision} \cdot \text{recall}}{	ext{precision} + \text{recall}}) \). The SVM-struct software package (Thorsten Joachims, 2007) was revised for this experiment.

4.2 Experiment Results

The initial experimental results are compared with the results using pair wise multi-class SVM reported in (Jun’ichi Kazama et al., 2002).

<table>
<thead>
<tr>
<th>Approach</th>
<th>Examples</th>
<th>Features</th>
<th>Precision</th>
<th>Recall</th>
<th>f-score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pair wise multi-class SVM</td>
<td>16,000 words</td>
<td>Words, pos, prefix, suffix,</td>
<td></td>
<td></td>
<td>36.82%</td>
</tr>
<tr>
<td>( F_{\text{basic}}(x_i, y_i; \omega) ) SVM-struct</td>
<td>16,000 words (586 sentences)</td>
<td>Words, pos, prec. class</td>
<td>51.00%</td>
<td>31.57%</td>
<td>37.53%</td>
</tr>
<tr>
<td>( F_{\text{hierarchical}}(x_i, y_i; \omega) ) SVM-struct</td>
<td>16,000 words (586 sentences)</td>
<td>Words, pos, prec. class</td>
<td>46.54%</td>
<td>31.73%</td>
<td>37.74%</td>
</tr>
</tbody>
</table>

Table 1. Experimental Results (best score of precision, recall, F-score separately).

From table 1, it can be found that the SVM-struct produces better F-score when compared with the pair wise multi-class SVM even though fewer features were employed by the SVM-struct. In addition, the SVM-struct with \( F_{\text{basic}}(x_i, y_i; \omega) \) which does not utilize taxonomy knowledge achieves better precision than the SVM-struct with \( F_{\text{hierarchical}}(x_i, y_i; \omega) \) which utilizes domain knowledge. However, \( F_{\text{hierarchical}}(x_i, y_i; \omega) \) leads to slightly better recall and F-score. The better recall achieved by \( F_{\text{hierarchical}}(x_i, y_i; \omega) \) can be explained in that the taxonomy knowledge empowers the classifier with better generalization capability such that more instances can be identified and classified.

5 CONCLUSION AND FUTURE WORK

We propose a novel ontology population method underpinned by the SVM-struct algorithm. Three kinds of discriminant functions based on the SVM-struct algorithm are examined. In particular, taxonomy knowledge captured in a domain ontology is used by the proposed ontology population method. The initial experimental results show that this ontology population method is more effective than the pair wise multi-class SVM method which has been used for ontology population tasks before. Our research work opens the door to the development of more effective automatic ontology extraction method for business knowledge acquisition. More comprehensive experiments will be conducted to
evaluate the proposed ontology population method in the future. For instance, our algorithm can be compared with other learning algorithms such as HMM (Lawrence R et al. 1998), CRF (John Lafferty et al., 2001), and Maximum Entropy (Adam L. et al., 1996). In addition, a customized kernel will be developed to take into account more features such as prefix, suffix, synonymous, syntactic relations, and semantic relations.

References


